Fairness Problems at the Media Access Level for High-Speed Networks *

Kurt Maly
L. Zhang D. Game
Department of Computer Science
Old Dominion University
Norfolk, VA 23529-0162

March 23, 1990

Abstract

Most lower speed ($\sim$ 10 Mb/s) local area networks use adaptive or random access protocols like Ethernet. Others at higher speed use demand assignment like token or slotted rings. These include Cambridge ring and electronic token ring systems. In this paper we discuss fairness issues in representatives of such protocols. In particular we selected FDDI as a demand access protocol using tokens, CSMA/RN a random access protocol and DQDB a demand access protocol using reservations. We focus on fairness at the media access level, i.e., attaining access or being excessively delayed when a message is queued to be sent as a function of network location. Within that framework we observe the essential fairness of FDDI, severe fairness problems in DQDB and some problems for CSMA/RN. We investigate several modifications and show their ameliorative effect. Finally we give a unified presentation which allows comparisons of the three protocols' fairness when normalized to their capacity.

*This work was supported by CIT grant RF-89-002-01, NASA grant NAG-1-908 and Sun Microsystems grant RF 596043.

1 Introduction

For the last ten years local area networks protocols have been dominated by Ethernet [9,1,14] and various token ring protocols with a nominal performance on the order of 10Mb/s. In 1990 implementation of metropolitan area networks are coming on the market with a performance on the order of 100Mb/s. For instance, FDDI [17,11,2,16] chipsets are now available with several vendors integrating them to make FDDI accessible with their products. FDDI is a token ring protocol designed to run at a speed of 100Mb/s. DQDB [10] is somewhat behind FDDI in reaching the market. It has been developed as a 300 Mb/s dual bus protocol (each bus is capable of handling 150Mb/s).

If predictions made in the Federal High Performance Computing Program [18,4] come true we will see within ten years operational gigabit widearea networks which will give individual users access to gigabit bandwidth while being separated thousands of kilometers. In a way we will have expanded local area networks from 10Mb/s to 1Gb/s and from few kilometers to thousands of kilometers.

The data rates mentioned above are realized at the lowest levels of the ISO model of communications. Currently the most commonly used protocol suit for higher levels - TCP/IP - reduces the effective speed by a factor of up to 10 in Ethernet. Most performance studies have been done at the medium access level. Considerably less information is available on the impact of overhead in layers above the transport layer. As new standards - OSI - are being developed to improve inter-operability of different computers, fears are expressed that the impact might prove too costly to be useful for gigabit networks. No matter what the future will bring for the higher levels it is absolutely essential that the media access protocol level be as efficient as possible. In this paper we shall concentrate on performance problems at the media access level and what, if anything, can be done about these problems at that level.

Traditional performance metrics for these protocols include delay metrics such as wait time for a message at a node, service time for a message, total response time (the time a message arrives at the destination minus the time a message arrives at a node), and throughput metrics such as bits delivered versus bits offered as a function of the load and the one we wish to concentrate on: fairness metrics.

Fairness in itself has many facets although in general it means that an entity should not have an advantage (as measured by some metric) over another entity. An entity may be a node, as for instance the node located physically at the end of the network should not have shorter access delays than a node in the middle of the network. An entity may be traffic of a specific type such as small messages versus large messages, or synchronous versus asynchronous. An entity may be load such as a node having to serve a large amount of traffic versus a node with little traffic and messages from either node should have the same access delay. In most cases fairness is only a problem over small time intervals. Most any protocol is fair if its behavior is observed over a long time period and statistics are calculated over that time period. On the other hand a faculty member editing a file remotely who gets a 30 second response time to a key stroke will be upset if his colleague at another node has a millisecond response time.

The general concept of fairness has been an issue in media access protocols since the beginning of local area networking. In carrier sensed systems, persistence, binary backoff, limited contention and a host of other techniques have been used to improve throughput and control access fairness [19]. Collision free protocols using basic bit mapping provide another method for each station to reserve a frame during the next transmission period. Since basic bit mapping is inherently unfair to the lower numbered nodes, BRAM, BRAP and other protocols were devised to alleviate the unfairness. [20]

Since, in contention protocols, there is the probability, however, remote, that a node's wait for access may be unbounded, token protocols were devised. Token protocols may maintain a priority system based upon actual token rotation time so that when heavily loaded, the ring will give higher priority messages better access. Later discussions of FDDI will examine token rotation access conditions in greater detail. Reference [6] has demonstrated that over time all stations on an FDDI ring have equal access to transmit asynchronous frames, regardless of the size of allocated synchronous bandwith for individual stations.

Slotted rings [21] provide control mechanism to establish access fairness. Each slot may contain access priority information or access to slots may be organized into rounds generated by a master station. Further access control can be based upon the fact that nodes must pass slots they empty to succeeding stations, or nodes may occupy only a single slot on the ring at any one time. In addition, some slotted rings provide different slot types, channel and normal, which can and cannot be reused by the emptying node, respectively. Unidirectional bus systems provide a number of access mechanisms including those based upon trains, cyclic polling and non-cyclic reservation schemes [21]. In the paper, we will examine the fairness for one particular reservation scheme, DQDB, in much greater detail.

In addition to fairness controls at the media access level, some fairness issues may be resolved by other means. For example, a node which needs guaranteed access may designate the information as synchronous, thus providing regular interval access in an integrated network. This may result in considerably wasted throughput and/or increased response time in the case where the data being generated by the node is highly dynamic and should normally be delivered asynchonously.

This leads to various concepts of fairness which are not compatible with each other. When transmitting voice it is necessary that a certain amount of information be transmitted regularly. If necessary other nodes may need to be starved in order that the rate for voice traffic be sustained. Here an absolute level of fairness is not the goal, but one should be able to specify a level of fairness to allow both synchronous traffic and a reasonable amount of asynchronous traffic. The remainder of the paper discusses how various protocols fare from the fairness point of view.

In section two we describe three representative protocols and their handling of fairness. We have selected FDDI as a representative of demand access protocols using tokens and DQDB as a representative of a demand access protocol using reservations both serving metropolitan area networks. The third one presented is, CSMA/RN, a protocol developed for gigabit wide area networks [3]. It is representative of random access protocols which normally have unbounded access delays. It is known that DQDB has the problem that wait time is a function of node location and we present several strategies to improve performance in this regard. Section three provides a summary of the results and also gives a comparison of the protocols involving the three major metrics: delay, throughput and fairness.

2 Fairness in three protocols

Fairness problems normally are associated with non-uniform load distributions across the nodes is a network. A solution to this problem is load balancing which attempts to match resources (bandwidth) with needs (offered load) at individual nodes. In order for this feature to handle sudden bursts, the reallocation of resources has to be done within milliseconds to be truly useful. One protocol which does so is DRAMA, [13,12,8,7], a protocol proposed for metropolitan areas. In none of the three protocols selected is load balancing fairness handled at the medium access level. We are currently in the process of developing distributed algorithms for FDDI, DQDB and CSMA/RN along the lines described in DRAMA, that is, each node, or group of nodes, keeps track of the overall network load and adjusts its own resources demands (bandwidth share) of the network accordingly. In future reports we shall describe our results in that direction.

All three protocols allow for different types of traffic, i.e., both synchronous and asynchronous traffic can be sent. Control of the allocation of bandwidth to nodes is handled in all cases at higher level. Problems with FDDI in handling isochronous traffic, i.e., guaranteed bandwidth at fixed intervals, have led to FDDI-II which we do not discuss here.

The problem we have studied in detail is the fairness to individual nodes compared to other nodes as measured by wait time and throughput even with a uniform load distribution across the network. To that end we have modeled all three protocols at the bit operational level. The models were implemented in Simscript for FDDI and DQDB and a mixture of C and Simscript for CSMA/RN. The details of the models and their validation can be found in [5,3].

Although we have made extensive studies of each protocol, we have selected for this paper only one case which illustrates the fairness problems we have found in these studies. The case selected is a 50 km long network with 50 equally spaced nodes. The traffic offered ranged from 0% - 100% for DQDB and FDDI and from 0% - 225% for CSMA/RN. Messages were 5000 bits long (with 10% variation) and were Poisson distributed; load was uniformly distributed among all nodes. For each protocol we will give a set of figures depicting wait time per message averaged over the entire network in μsec and throughput measured in percent of capacity as a function of offered load respectively. For each of these two curves we will give two or more snapshots, taken at various levels of load, of the same metrics averaged for each node and displayed for each node.

2.1 FDDI

FDDI is a token ring protocol. In Figure 1 a message is on the ring with the token appended to its end. Node D is the next one with a message and, ignoring the token rotation time for the moment, it will pick up the token and send its message as shown in Figure 2. Fair access and throughtput for individual nodes is obtained by means of a token rotation time (TRT) which guarantees that on the average each node will have access to the ring within TRT. In the worst case a node may have to wait 2*TRT. This, however, does not guarantee useful access to the ring in that time period. A node may get the token but no time may be left to send a message. In the worst case - all nodes having messages queued up - the token ring protocol will give each node in a round-robin way useful time to send messages although it may take $(n-1)$ *TRT where n is the number of nodes. For a detailed discussion see [6].

Figure 3 confirms the essential fairness of FDDI; at 80% the variation from the mean wait time of the *net* is *about 10% for individual* nodes. At higher loads the absolute amount of the variation increases significantly but the percentage grows much slower. For instance, at 90% the variation in wait time is 5,315 μsec with the net average being 22,309 μsec. As indicated earlier though, synchronous bandwidth is not handled at this level and it can happen that with nodal distribution varying over time that access delays will vary as a function of nodes. Secondly the variation in TRT does not allow for isochronous traffic. Thirdly the establishment of an effective TRT for a particular environment is not a trivial issue [5]. In this discussion we have ignored the fact that FDDI actually consists of dual counter-rotating rings for fault tolerant purposes because this does not effect the issue of fairness presented here.

2.2 CSMA/RN

CSMA/RN [3] is an extremely simple - hence it should prove inexpensive to implement - protocol which relies on the fact that a network looks quite differently at high speed ($>$ 100 Mb) than at low speed ($\sim$ 10 Mb). In the latter case only a fraction of one message occupies the entire ring while in the former case several messages can be on the ring, that is, if we assume uni-directional transmissions.

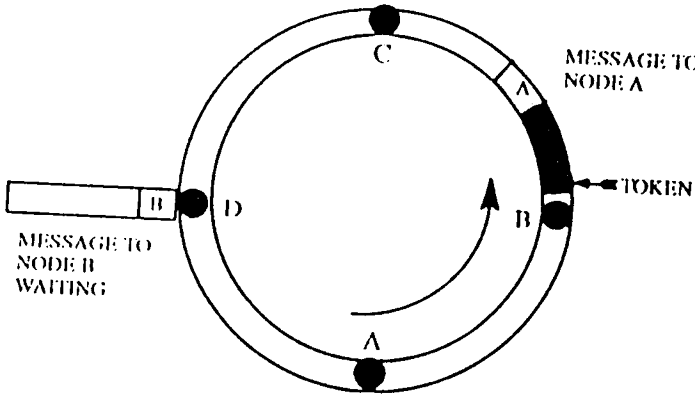


Figure 1: Message on token ring

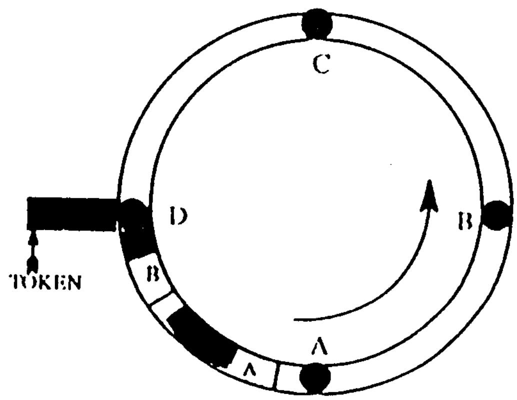


Figure 2: Node D captures token and sends its message.

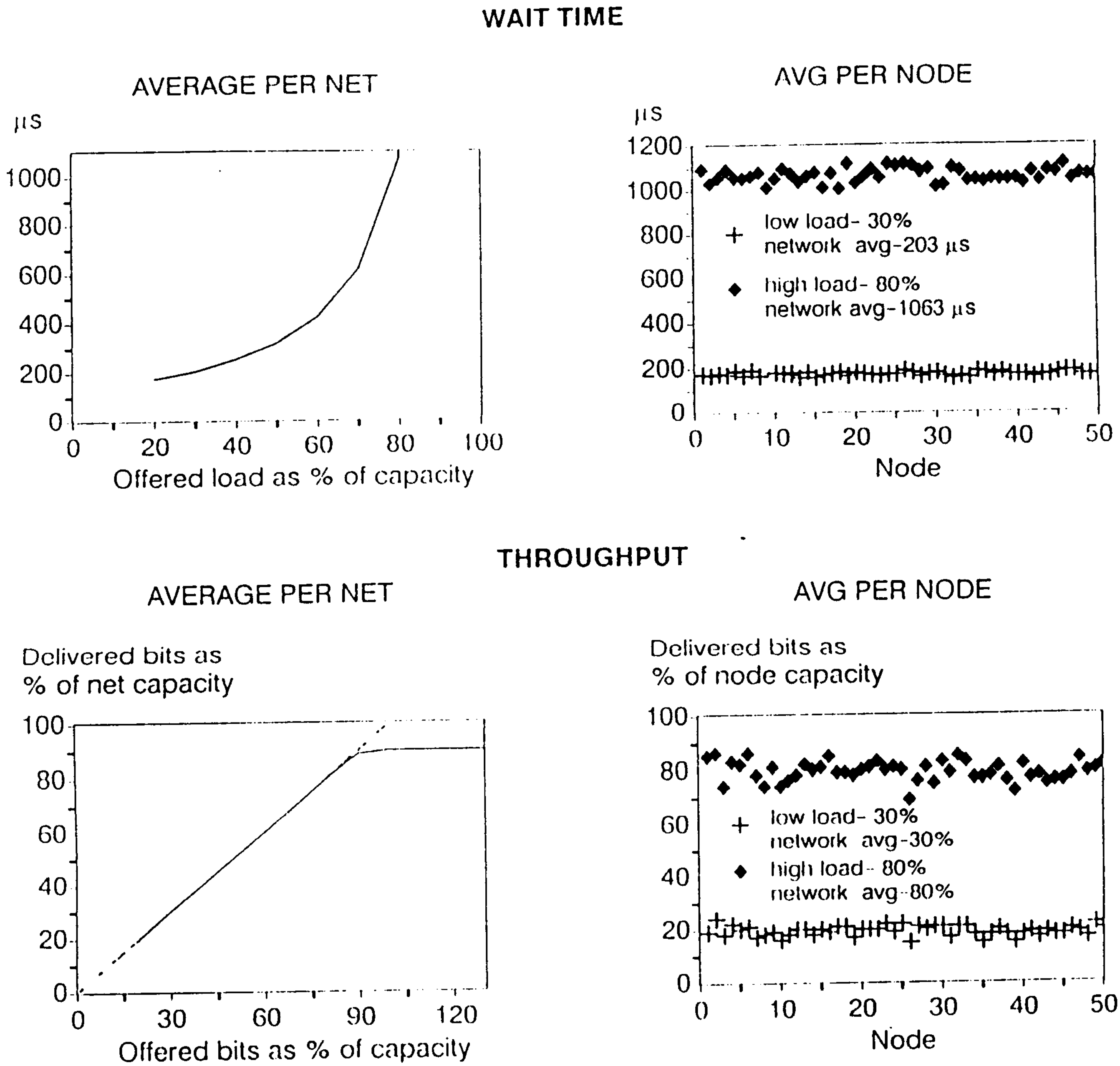


Figure 3: Fairness graph for FDDI

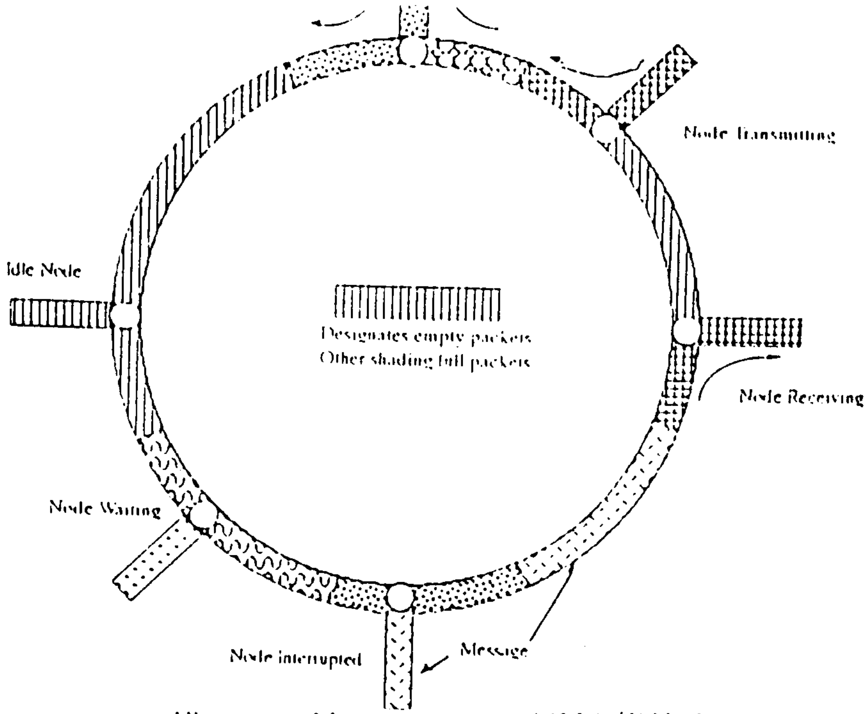


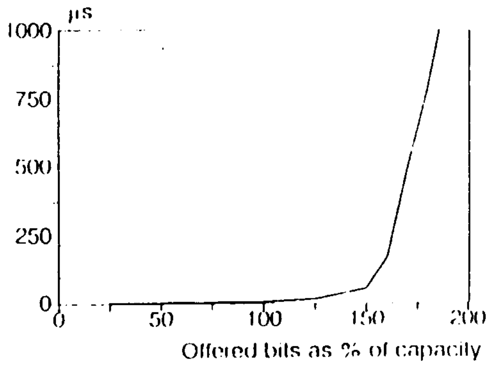
Figure 4: Messages on a CSMA/RN ring

The protocol is as follows: if a node has a message to send it does so if it does not detect another message passing by. If the ring is busy the node defers until it detects an idle ring at which time the node starts sending. If a sending node detects an incoming message whose destination is itself, the node continues sending while it takes the incoming message off the ring. If the incoming message is for another node, the sending node interrupts the message it is sending and lets the incoming message pass by. The remainder of the interrupted message is then treated as a new message. This implies that a receiver may have to reassemble the fractured messages. Figure 4 depicts various states the ring can be in. In [3] a detailed analysis of this protocol shows a truly outstanding performance over a wide range of parameter choices. For instance, with boards capable of handling one gigabit/s at each node, the net can actually deliver a throughput of two gigabits/s.

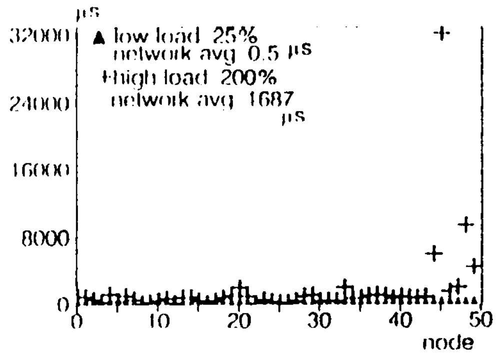
Fairness in most cases is quite good for up to 150% of offered load at 1Gbps (see Figure 5) but problems do occur. At higher loads the variation of the nodes' averages is beginning to increase. In specific cases actual starvation of some of the nodes can occur when they are unable to achieve their share of throughput even at higher wait times. Which nodes exhibit starvation does not depend on the node's physical location but who sends what message to who. Also this phenomenon is exhibited by some nodes for a short duration and starvation keeps drifting from node to node on the ring

WAIT TIME

AVERAGE PER NET

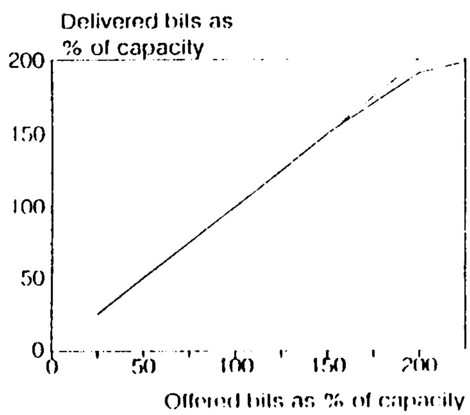


AVG PER NODE



THROUGHPUT

AVERAGE PER NET



AVG PER NODE

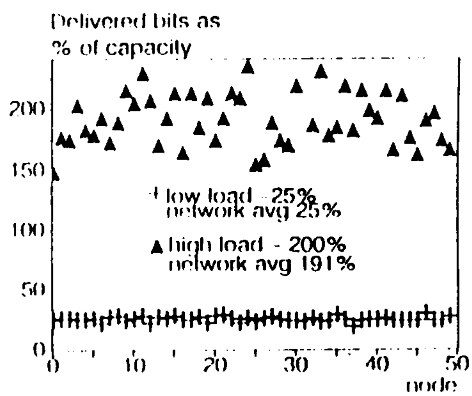


Figure 5: Fairness Graphs for CSMA/RN at 1Gb/s

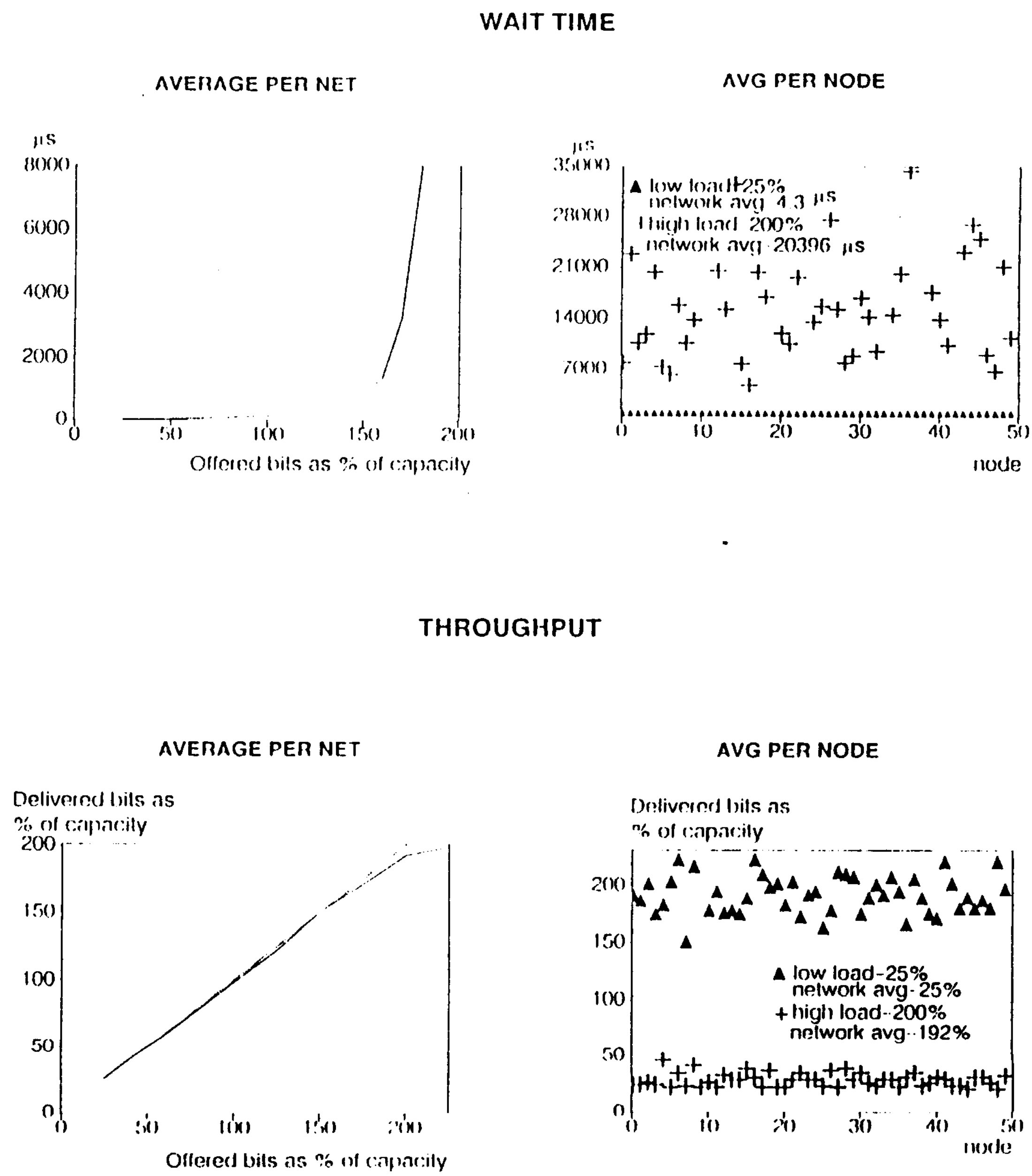


Figure 6: Fairness Graphs for CSMA/RN at 100 Mb/s

along the course of time. Not visible in the graph is the information that till 150% load no node starves, at 200% load four nodes suffer severe starvation sending from 166% to 176% and at 225% load nodes had throughput between (90% - 200%), twelve nodes had between 180%-190% and five nodes had below 180% throughput. Since the results are based upon random distribution, conditions will vary over different runs and over different intervals in a run.

In Figure 6 we have run CSMA/RN at the speed of FDDI - 100Mbps. As expected CSMA/RN performance has deteriorated although it is still better than FDDI in terms of throughput and wait time metrics. The point of instability lies between 150% and 175% of offered load instead of being beyond 200%. The starving node phenomenon is observed at 100% offered load and it deteriorates at higher loads. At 200% load sixteen nodes are severely affected and the variance of wait time and throughput for nodes is even higher than CSMA/RN operation at 1Gbps.

Research is ongoing to ensure that no node starves in this protocol. As indicated earlier, we are developing automatic load balancing algorithms at the media access level to solve unfairness problems due to non-uniform load distribution. As a side effect we expect that these algorithms will also solve the fairness problems for uniform load distribution.

2.3 DQDB

DQDB [10] uses a dual slotted bus to send segments (messages are partitioned and reassembled by the protocol) of 312 bits. A node uses one bus to send downstream and the other bus to send upstreams. Reservation bits are sent in the opposite direction to alert nodes to let empty segments go by to the nodes making reservations. Since the system is totally symmetric we shall illustrate it with sending in one direction only.

Each node has a reservation bit counter which keeps track of how many requests have been made downstream. If a segment arrives at a node the segment gets stamped with the counter value, say x, and the counter is reset to zero. When x empty slots have passed, this segment is sent in the next empty slot.

When a segment arrives, the node attempts to send its reservation upstream. DQDB uses one bit of the 312 bit segment as the reservation bit, hence the node has to wait till a segment with a free reservation bit arrives. Therefore, it is possible that the segment has been sent already while the reservation bit is still waiting. As a consequence reservation bits can queue up while segments are actually being sent.

In Figure 7 we have shown a particular instance of DQDB. A segment with the destination for node B arrives at node A. Since two nodes have already made reservations (reservation bit counter = 2), the stamp is set to 2, and node A will be able to use the third empty segment, labeled 'a'. In the mean time, the reservation will have to wait till the old one (attempted reservation = 1) gets reservation slot 'b' after which node A will send its reservation in slot labeled 'c'. Thus, the segment will have been sent before a reservation was made. At this point the reservation bit counter will be up to a value of 3.

Under normal circumstances, DQDB will order all segments in the network waiting to be sent if we neglect the time it takes for signals to reach the other nodes. However, that does not apply to nodes where queues of segments (and messages) have built up. Each queue will be ordered and all the front segments of the queues will be ordered but elements in two different queues are not necessarily ordered with regard to each other. We shall denote this case as the *non-ordered* one.

We have developed models with different strategies for when to send the reservation bit and whether or not to fully order messages (and segments). The first strategy for sending the reservation bit is as described above and we shall refer to it as the *aggressive* one. In the second strategy we delay the stamping of the front segment until the reservation bit has been sent. We refer to this strategy as the *non-aggressive* one. A third strategy, the *combined* one, uses a Bernouilli trial to decide what strategy to employ for each message (not segment).

We also investigated the impact of fully ordering segments across all queues. In the ordered case, we attach to the state of the reservation bit counter each segment as it enters the queue and reset the counter to zero. When the front segment is stamped to start the count down, it is from this field that the value for the stamp is taken.

Figure 8-12 are the results of running the six combinations of reservation bit strategies and ordering across queues. Since we show these combinations for only one set of parameters, a short note on the impact of other parameters on DQDB's performance is appropriate. In the aggressive case, the wait time is strictly a function of how many reservation bits have come from downstream and how many segments arrive at upstream nodes before this

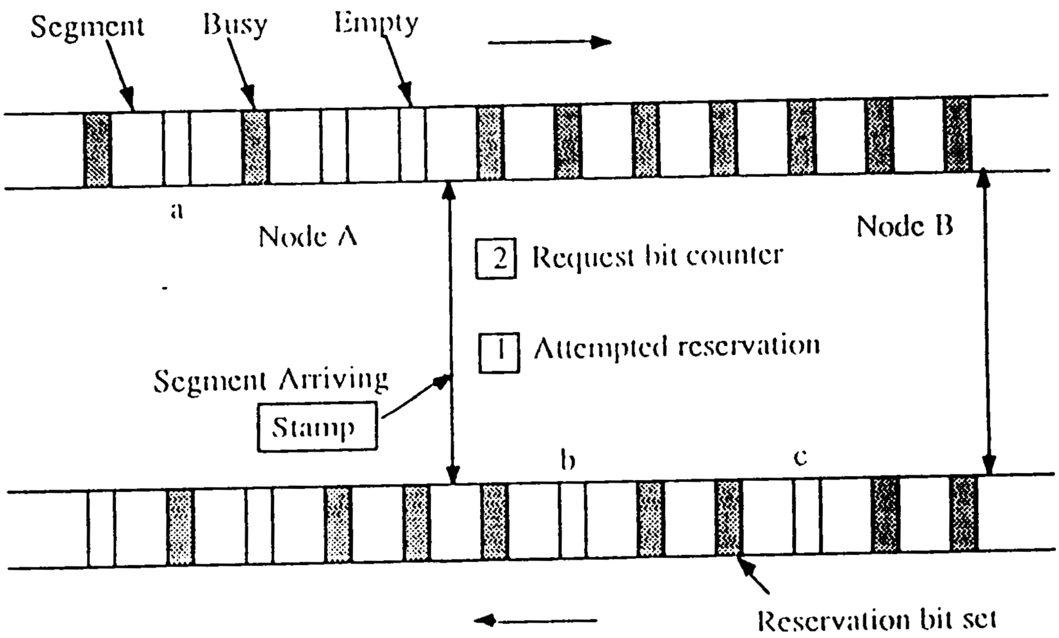


Figure 7: Instance of DQDB

segment's arrival can be signaled to those nodes. This, in turn, is a function of the number of nodes which want to send downstream and what their load is. In the non-aggressive case, the wait time is more directly tied to the availability of empty reservation bit slots coming from the downstream side. If enough nodes, have enough messages arriving, and are located within a short enough distance then it is possible to starve a node because no reservations bit will be empty.

For DQDB, we show the averages per node as a function of the node position and not just as a collection of discrete points. There clearly is a functional relation between node positions and node performance. We remind the reader that the wait time in the following figures are for messages and not for segments.

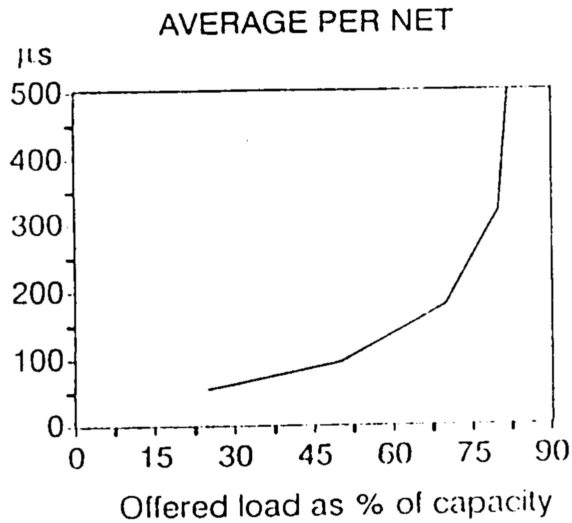
Figures 8 and 9 illustrate the impact that ordering has on aggressive DQDB. At 80% the curvature of the nodal curve flips from concave to convex; in unstable situations the curvature is concave in both cases. Ordering has also a marked effect on the network average. Comparing Figures 9 and 10 we can observe the effect of changing from an aggressive to a non-aggressive strategy. The non-aggressive strategy produces worse network averages, better performance for the middle nodes and worse performance at the end nodes. This holds true for both throughput and wait time. Figure 11 shows that a simple uniform combination does not flatten out the curve although it produces better performance for more nodes and the number of badly affected nodes is reduced. The cost for this improvement is a slightly worse network average.

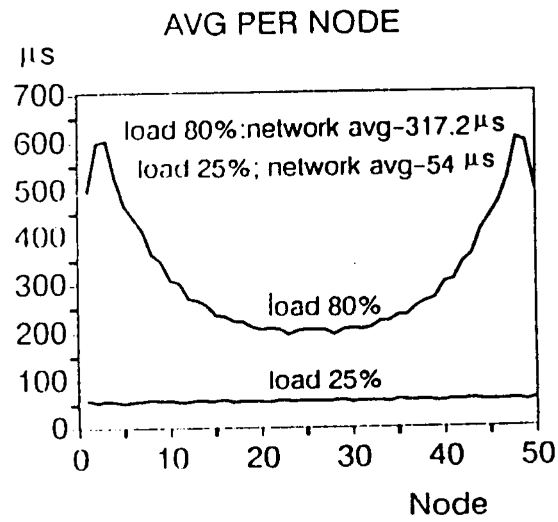
For completeness sake we give the two remaining non-ordered cases in Figures 12 and 13 noting that their performances are worse than their ordered counterparts.

Table 1 gives a comparison of the various cases for wait time averages for the entire net at 90%. It should be noted that in all cases the net is on the boundary of being overloaded and wait time per se is not a meaningful figure. Since we made the runs for the same length of simulation time we still can make meaningful relative comparisons.

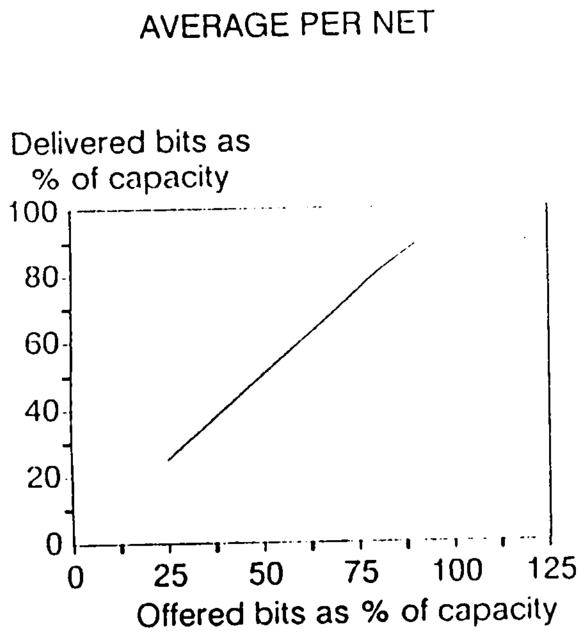
Figure 14 shows finally a flat curve or in other words a fair protocol. It is obtained by running a non-uniform combination strategy. At the endnodes the probability for using the aggressive strategy is .90 which is reduced parabolically down to .5 for most of the middle nodes. This type of com-

WAIT TIME





THROUGHPUT



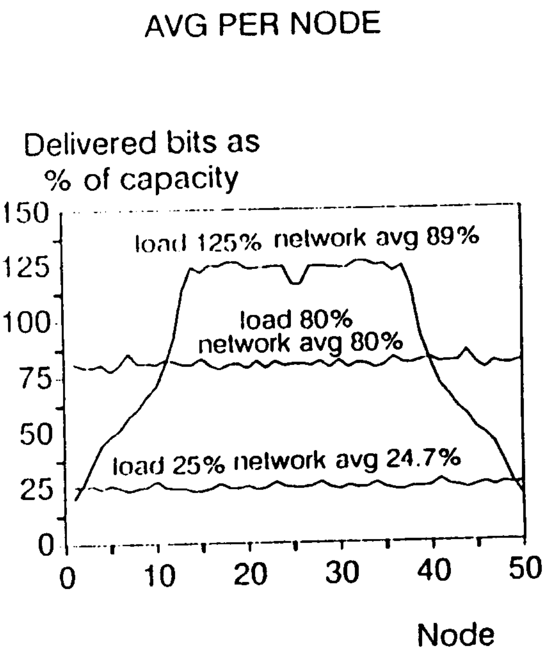


Figure 8: Fairness in non-ordered, aggressive DQDB

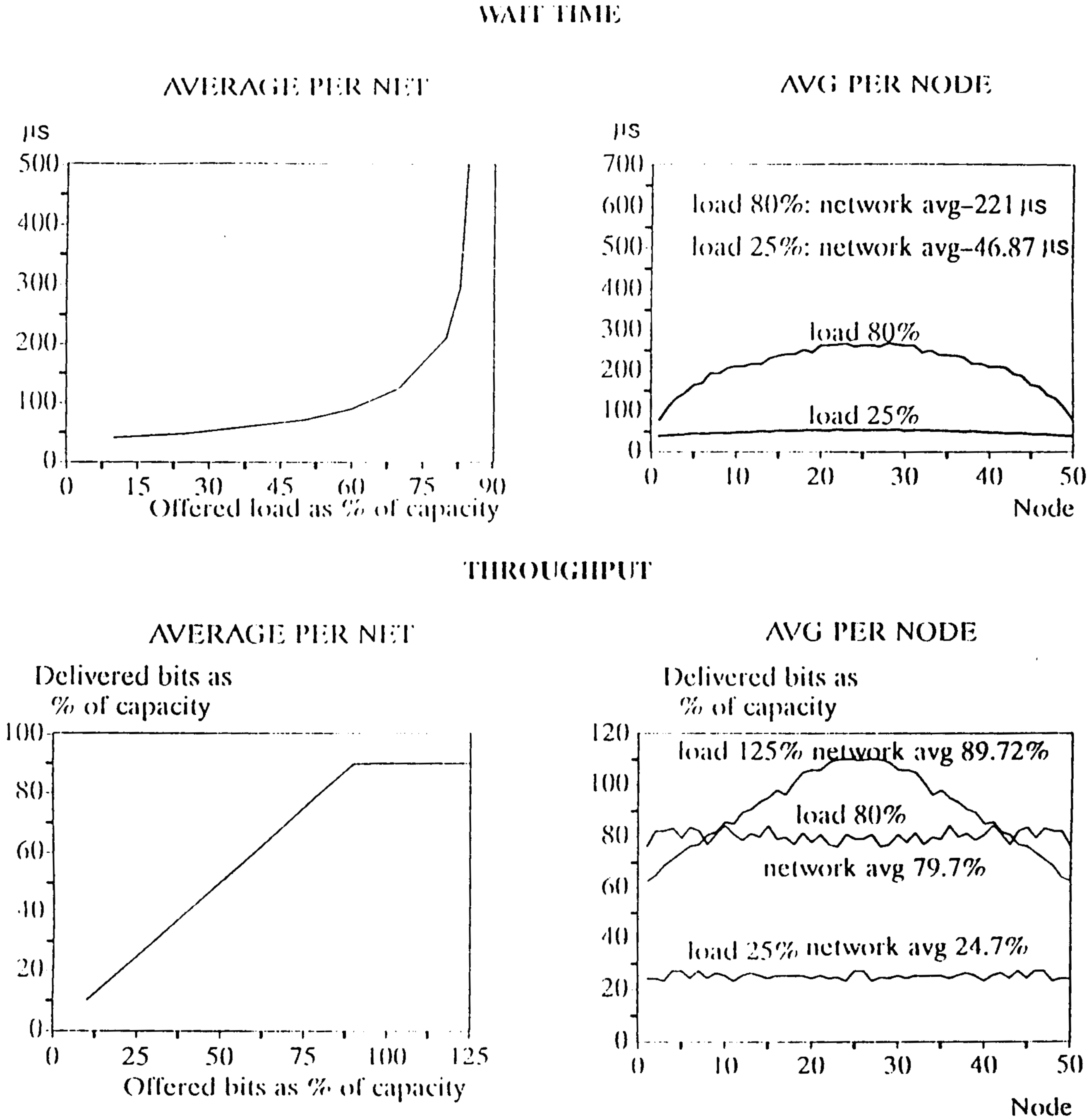


Figure 9: Fairness in ordered, aggressive DQDB

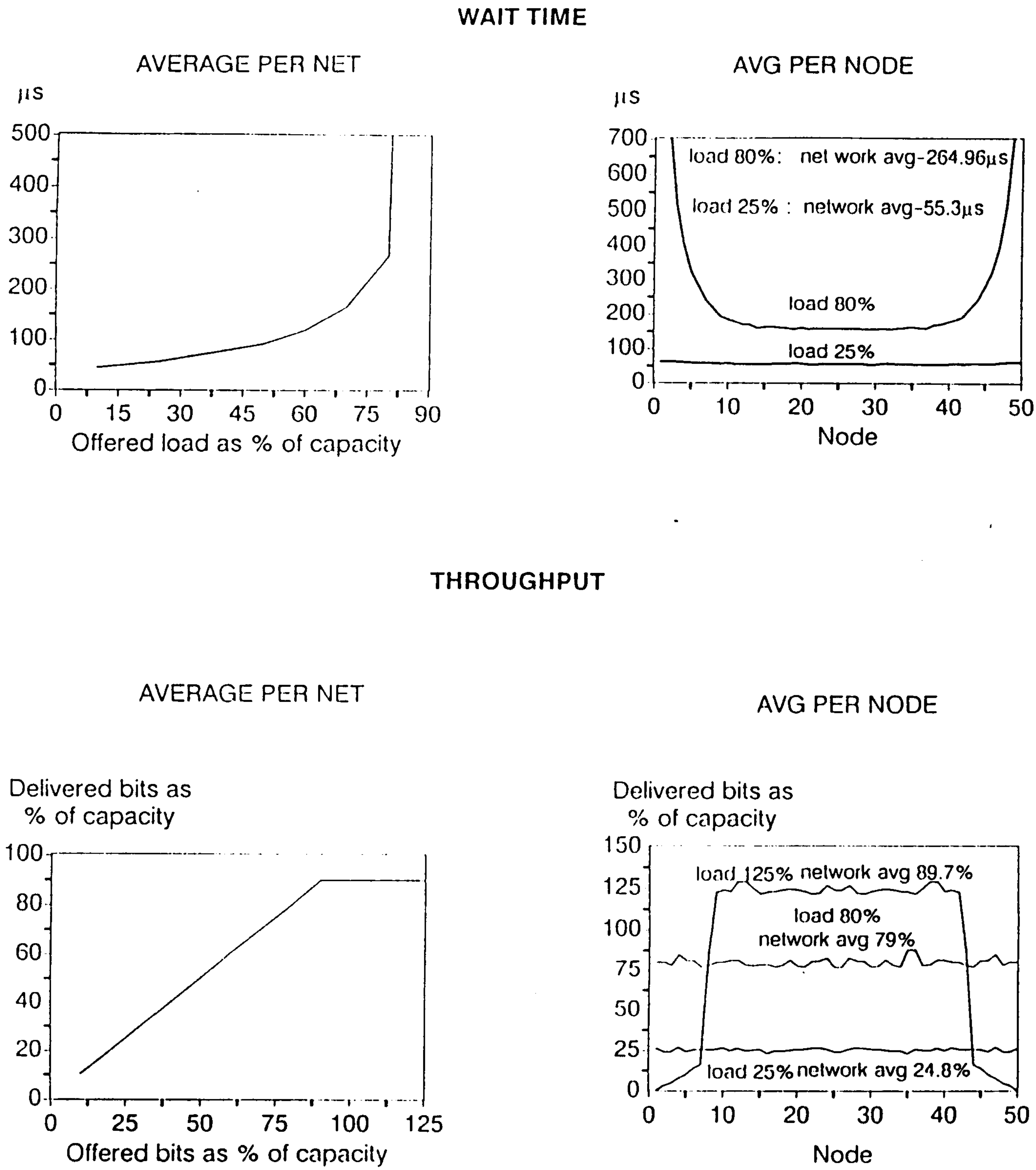


Figure 10: Fairness in ordered, non-aggressive DQDB

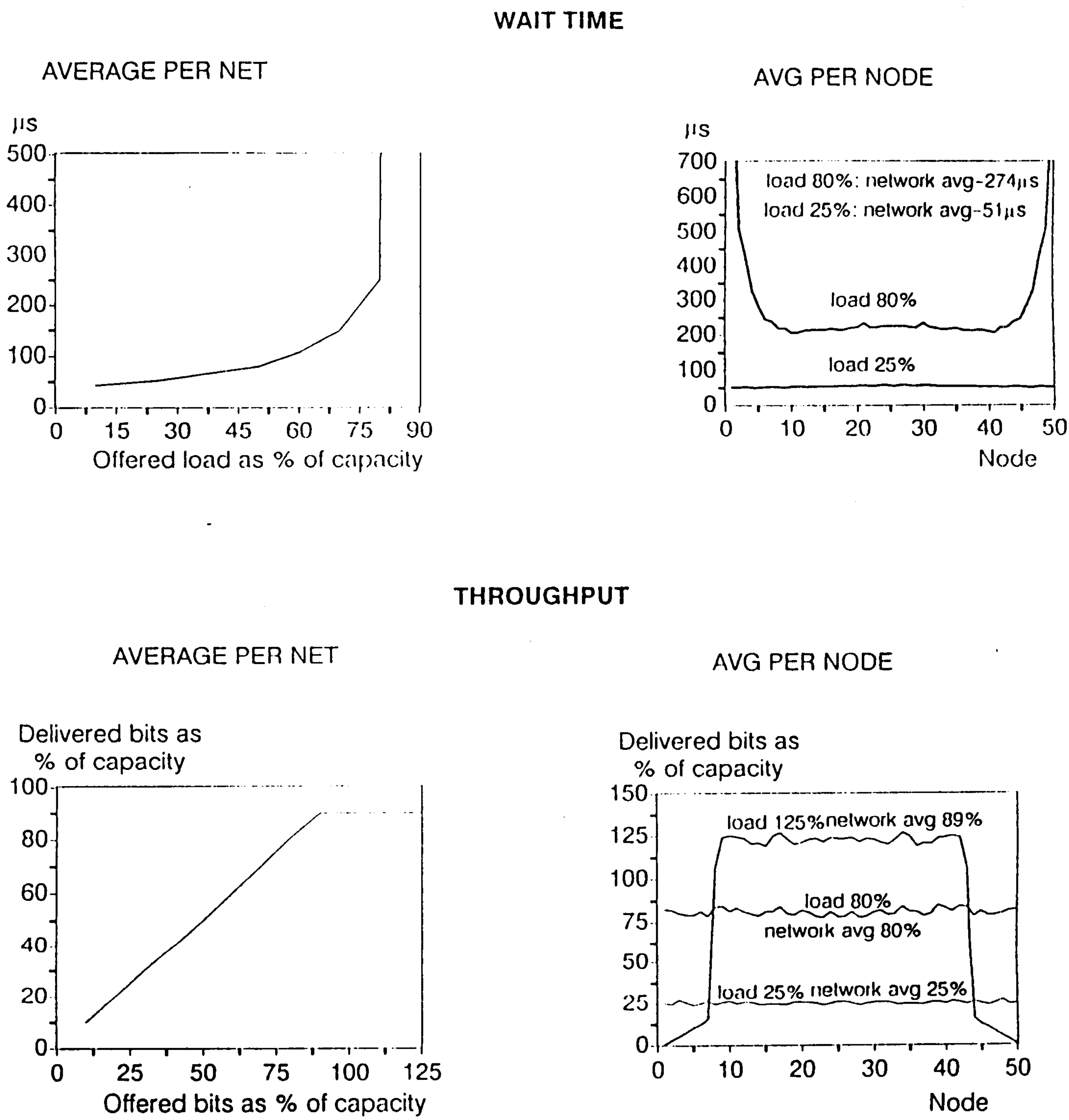


Figure 11: Fairness in ordered, combined DQDB

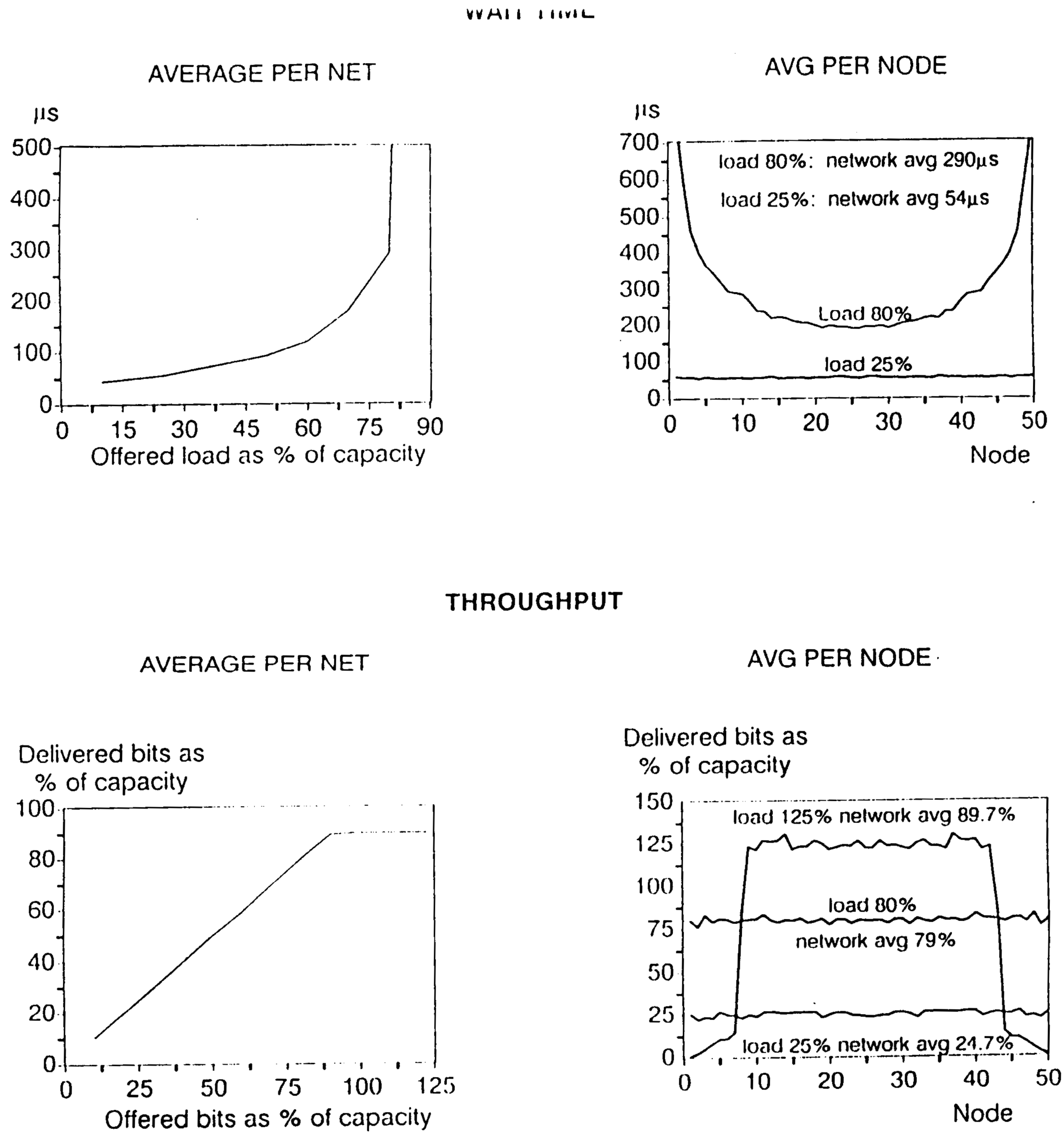
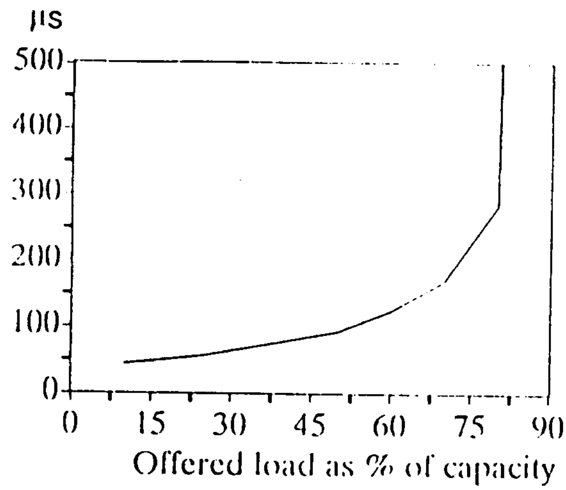


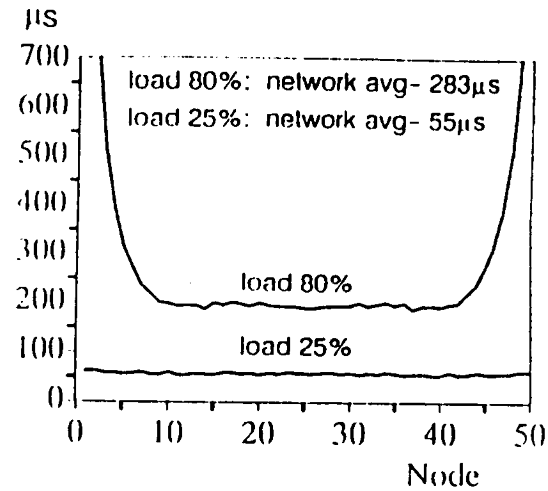
Figure 12: Fairness in non-ordered, combined DQDB

WALL TIME

AVERAGE PER NET

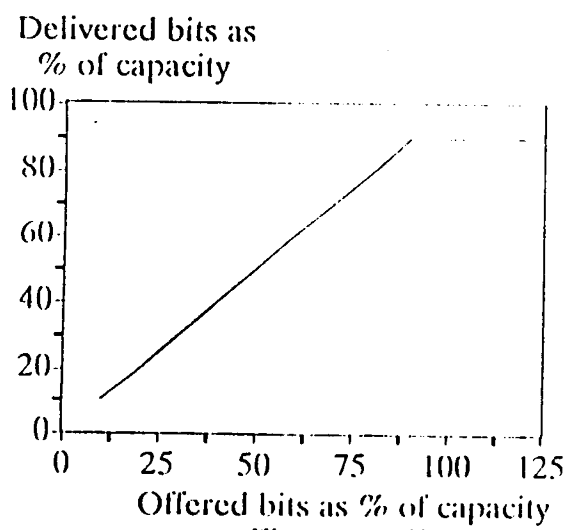


AVG PER NODE



THROUGHPUT

AVERAGE PER NET



AVG PER NODE

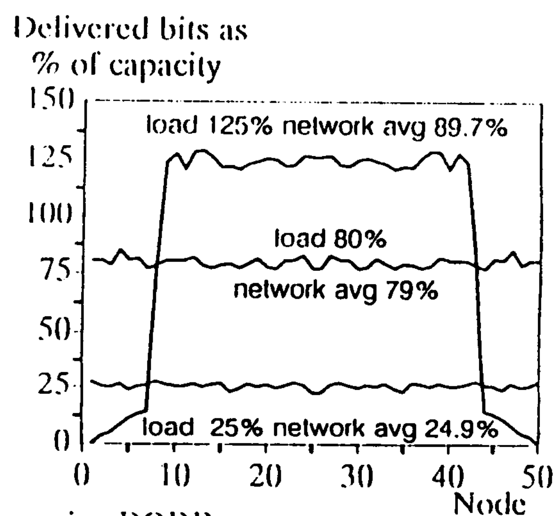
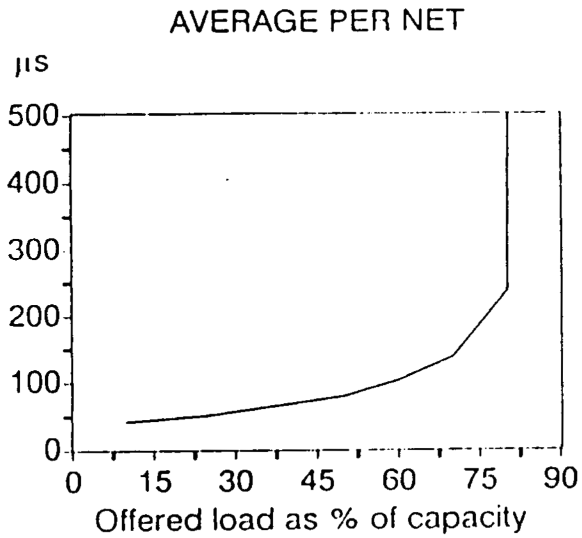
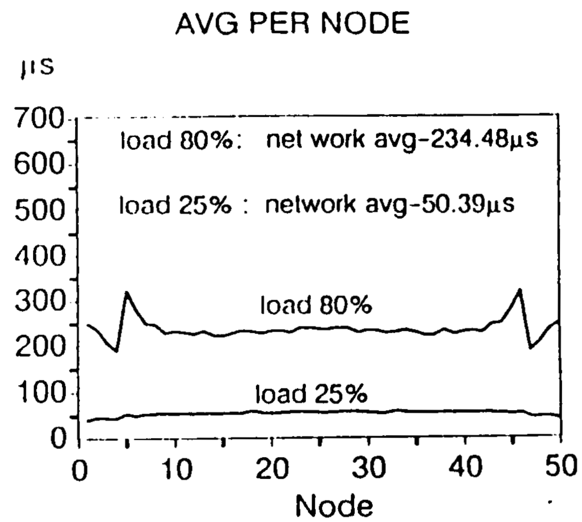


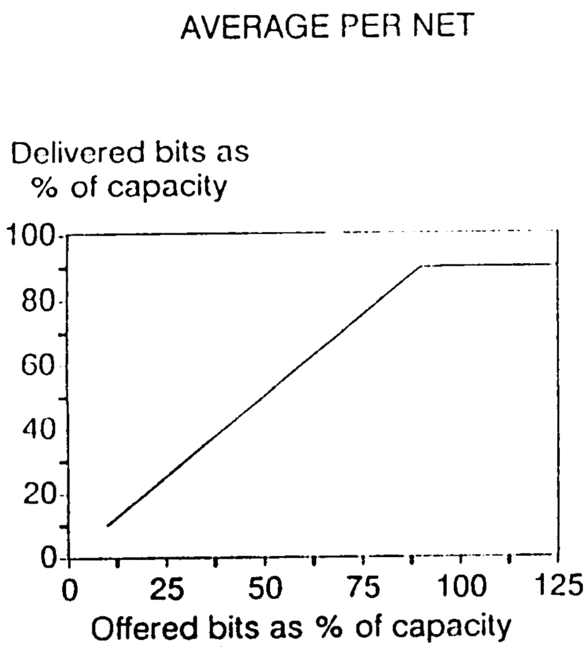
Figure 13: Fairness in non-ordered, non-aggressive DQDB

WAIT TIME





THROUGHPUT



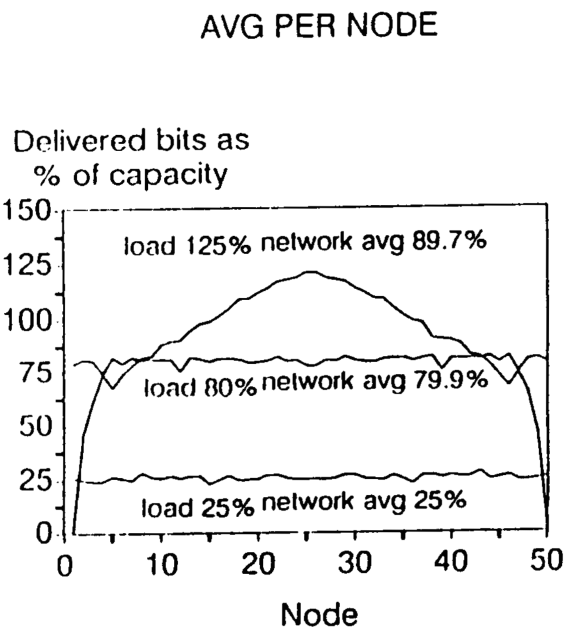


Figure 14: Fairness in ordered, non-uniform combined DQDB

Type of Network Model	Avg Wait Time
Non-ordered Non-aggressive	4440.06 μs
Non-ordered Aggressive	5310.6 μs
Non-ordered Combination	3091.56 μs
Fully Ordered Non-aggressive	4388.48 μs
Fully Ordered Aggressive	9843.61μs
Fully Ordered Combination	8858.03μs
Fully Ordered Non-uniform Combination	7200.7 μs

Table 1: Average Wait Time per Network at 90% Load

bined strategy produces the fairest of the ones investigated although even this combination produces unfair situations at overload. These results are convincing enough to make us believe that DQDB can be made fair at least to the degree FDDI is fair. The avenue we are pursuing, as with CSMA/RN, is to solve the problem of automatic load balancing for non-uniform load at the cost of taking some bandwidth for global communication. This solution in combination with the approach above should enable us to make DQDB fair in all cases.

3 Comparisons and conclusions

FDDI develops random variations in performance across nodes at overload but the variations are well within acceptable limits (about 20%). CSMA/RN shows an increase in variations of the performance metrics as load increases. The variation becomes significant (greater than 100%) at loads greater than 150% for as much as 20% of the nodes and in extreme cases can lead to temporary starvation of nodes. The variations are not a functions of the location of a node but occur in random positions. DQDB shows unfairness at higher loads as a function of a nodes' position in the network. Depending on the strategy selected end nodes are better or worse off than middle nodes. One particular, non-uniform combination of an aggressive and a non-aggressive DQDB version produces fair performance up until overload.

Clearly, comparing the performance of a 100 Mb/s, a 300 Mb/s, and a Gb/s protocol is akin to comparing apples and oranges. But we wanted to see how the protocols compare in terms of fairness for a fixed data rate. Therefore, Figure 13 and 14 give first a comparison for the network averages of the three protocols followed with nodal snapshots at 80 Mb/s, 270 Mb/s and 2 G/bs. When viewed in isolation FDDI was the most trouble free but when compared with the other protocols at constant capacity it actually is the worst. Other factors such as reliability and cost though will change this picture for different environments and no one absolute statement can be made.

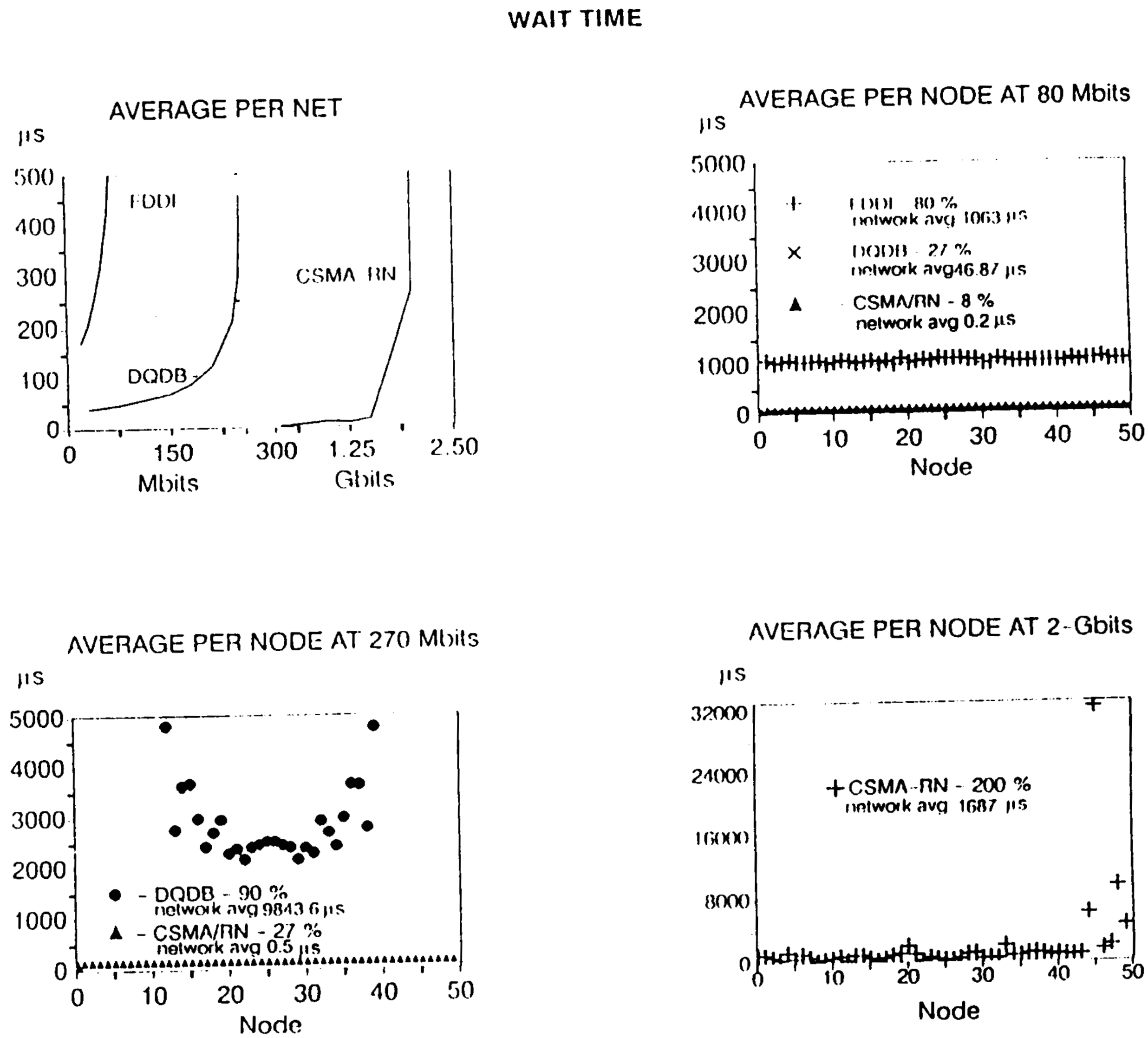


Figure 15: Comparison of access delays for FDDI, DQDB, and CSMA/RN

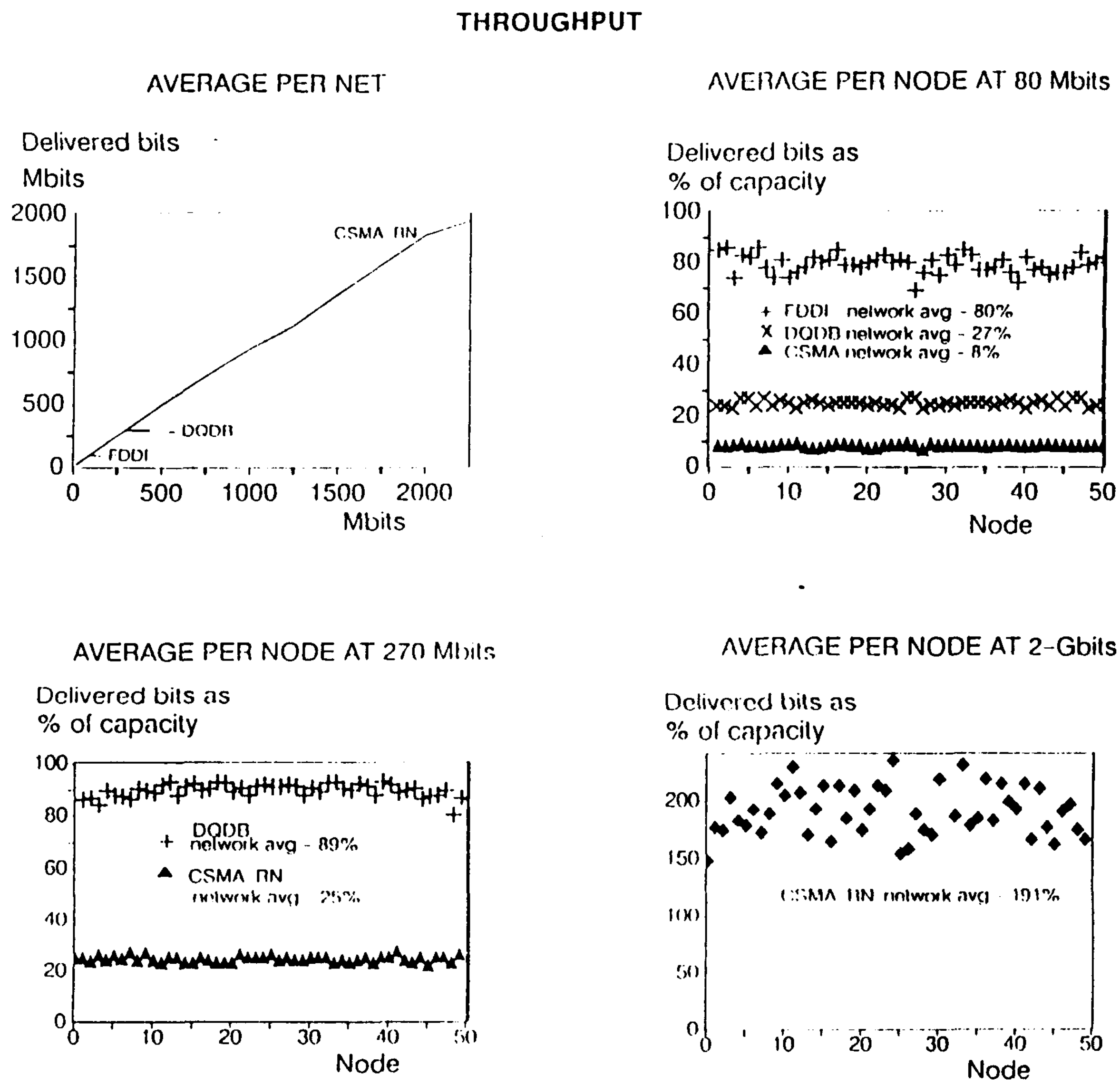


Figure 16: Comparison of throughputs for FDDI, DQDB and CSMA/RN

References

[1] D.R. Boggs, J.C. Mogul, and C.A. Kent. Measured capacity of an ethernet: myths and reality. *SIGCOMM Symposium*, 222–234, August 1988.

[2] W.E. Burr. The fddi optical data link. *IEEE Communications*, 24(5):18–23, May 1986.

[3] E. Foudriat, K. Maly, C.M. Overstreet, Sanjay Khanna, and Frank Paterra. *A Carrier Sensed Multiple Access Protocol for High Data Rate Rings*. Technical Report, Old Dominion University, Feb 1990.

[4] Peter Freeman. High performance computing a brief analysis. *Computing Research News*, 2, no.1:1,19, 1990.

[5] D. Game and K. Maly. *Extensibility and Feasibility of FDDI*. Technical Report, Old Dominion University, Feb 1990.

[6] M.J. Johnson. Proof that timing requirements of the fddi token ring protocol are satisfied. *IEEE Transactions on Communications*, 35(6):620–625, June 1987.

[7] K. Maly, E.C. Foudriat, D. Game, R. Mukkamala, and C.M. Overstreet. Traffic placement policies for a multi-band network. *SIGCOMM Symposium*, September 1989.

[8] K. Maly, C. M. Overstreet, X. Qiu, and D. Tang. Dynamic resource allocation in a metropolitan area network. *SIGCOMM Symposium*, 13–24, August 1988.

[9] R.M. Metcalfe and D.R. Boggs. Ethernet: distributed packet switching for local computer networks. *Commun. ACM*, 19, July 1976.

[10] R.M. Newman, Z.L. Budrikis, and J.L. Jullett. The qpsx man. *IEEE Communications Magazine*, 26(4):20–28, April 1988.

[11] F. Ross. Fddi - a tutorial. *IEEE Communications*, 24(5):10–17, May 1986.

[12] S. Sharrock, K. Maly, S. Ghanta, and H. Du. A broadband integrated voice/data/video network of multiple lans with dynamic bandwidth partitioning. *INFOCOM '87*, 417–425, March 1987.

[13] S. Sharrock, K. Maly, S. Ghanta, and H. Du. A framed, movable-boundary protocol for integrated voice/data in a lan. *SIGCOMM '86*, 9 Pages, August 1986.

[14] J. F. Shock and J. Hupp. Measured performance of an ethernet local network. *Communications of the ACM*, 711–721, December 1980.

[15] M. Skov. Implementation of physical and media access protocols for high-speed networks. *IEEE Communications Magazine*, 45–53, June 1989.

[16] Draft Proposed American National Standard. Fddi token ring media access control (mac) asc x3t9.5 rev. 10. February 28, 1986.

[17] S.L. Wallach. Fddi tutorial: lan industry gets another standard. *LAN Magazine*, 44–47, March 1987.

[18] Paul Young. Challenges for computing research. *Computing Research News*, 1, no.2:1–3, 1989.

[19] Jason S. J. Chen and Victor O. K. Li. Reservation CSMA/CD: A Multiple Access Protocol for LAN's. *IEEE Journal on Selected Areas in Communication*, 202–211, February 1989.

[20] Andrew S. Tanenbaum Computer Networks and Communication. *Chapter 3.*

[21] Morten Skov. Implementation of Physical and Media Access protocols for High Speed Networks. *IEEE Communications Magazine*, 45–53, June 1989.

[22] Mirjana Zafirovic-Vukotic, Ignas G. Niemegeers and Durk S. Valk. Performance Analysis of Slotted Ring Protocols in HSLAN's. *IEEE Journal on Selected Areas in Communication*, 1011–1024, July 1988.